Robust Low-Rank LCMV Beamforming Algorithms Based on Joint Iterative Optimization Strategies

Rodrigo C. de Lamare

***Abstract*—This chapter presents reduced-rank linearly constrained minimum variance (LCMV) algorithms based on the concept of joint iterative optimization of parameters. The proposed reduced-rank scheme is based on a constrained robust joint iterative optimization (RJIO) of parameters according to the minimum variance criterion. The robust optimization procedure adjusts the parameters of a rank-reduction matrix, a reduced-rank beamformer and the diagonal loading in an alternating manner. LCMV expressions are developed for the design of the rank-reduction matrix and the reduced-rank beamformer. Stochastic gradient and recursive least-squares adaptive algorithms are then devised for an efficient implementation of the RJIO robust beamforming technique. Simulations for a application in the presence of uncertainties show that the RJIO scheme and algorithms outperform in convergence and tracking performances existing algorithms while requiring a comparable complexity.**

***Index Terms*—Adaptive beamforming, constrained optimization, robust techniques, reduced-rank methods, iterative methods.**

I. INTRODUCTION

In the last decade, techniques have attracted a significant interest from researchers and engineers, and found applications in radar, sonar, wireless communications and seismology [1], [2]. The optimal linearly constrained minimum variance (LCMV) beamformer is designed in such a way that it minimizes the array output power while maintaining a constant response in the direction of a signal of interest (SoI) [1], [2], [3]. However, this technique requires the computation of the inverse of the input data covariance matrix and the knowledge of the array steering vector. Adaptive versions of the LCMV beamformer were subsequently reported with stochastic gradient (SG) [4], [5] and recursive least squares (RLS) [7] algorithms. A key problem with techniques is the impact of uncertainties which can result in a considerable performance degradation. These mismatches are caused by local scattering, imperfectly calibrated arrays, insufficient training and imprecisely known wave field propagation conditions [2].

In the last decades a number of robust approaches have been reported that address this problem [8]-[31]. These techniques can be classified according to the approach adopted to deal with the mismatches: techniques based on diagonal loading [8], [10], [13], [14], methods that estimate the mismatch or equivalently the actual steering vector [11], [12], [15], and techniques that exploit properties such as the constant modulus of the signals [16], [17], [18] and the low-rank of the interference subspace [9],[19]-[31]. Furthermore, beamforming algorithms usually have a trade-off between performance and computational complexity which depends on the designer's choice of the adaptation algorithm [3], [6]. A number of robust designs can be cast as optimization problems which end up in the so-called second-order cone (SOC) program, which can be solved with interior point methods and have a computational cost that is super cubic in the number of parameters of the beamformer. This poses a problem for beamforming systems that have a large number of parameter and operate in time-varying scenarios, which require the beamformer to be recomputed periodically.

A robust technique for short-data record scenarios is reduced-rank signal processing [19]-[31], which is very well suited for systems with a large number of parameters. These algorithms are robust against short data records, have the ability to exploit the low-rank nature of the signals encountered in beamforming applications and can resist moderate steering vector mismatches. These methods include the computationally expensive eigen-decomposition techniques [19]-[20] to alternative approaches such as the auxiliary-vector filter (AVF) [21],[26], the multistage Wiener filter (MSWF) [22], [24], [25] which are based on the Krylov subspace, and joint iterative optimization (JIO) approaches [23], [27], [28], [30], [29]. The JIO techniques reported in [27], [28], [30] outperform the eigen-decomposition- and Krylov-based methods and are amenable to efficient adaptive implementations. However, robust versions of JIO methods have not been considered so far.

In this chapter, robust LCMV reduced-rank beamforming algorithms based on constrained robust joint iterative optimization (RJIO) of parameters are developed. The basic idea of the RJIO approach is to design a bank of robust adaptive beamformers which is responsible for performing dimensionality reduction, whereas the robust reduced-rank beamformer effectively forms the beam in the direction of the SoI and takes into account the uncertainty. Robust LCMV expressions for the design of the rank reduction matrix and the reduced-rank beamformer are proposed that can appropriately deal with array steering vector mismatches. SG and RLS algorithms for efficiently implementing the method are then devised. An automatic rank adaptation algorithm for determining the most adequate rank for the RJIO algorithms is described. A simulation study of the proposed RJIO algorithms and existing techniques is considered.

This chapter is organized as follows. The system and signals models are described in Section II. The full-rank and the reduced-rank LCMV filtering problems are formulated in

R. C. de Lamare is with the Communications Research Group, Department of Electronics, University of York, York Y010 5DD, United Kingdom . E-mail: rcdl500@ohm.york.ac.uk

Section III. Section IV is dedicated to the RJIO method, whereas Section V is devoted to the derivation of the adaptive SG and RLS algorithms, the analysis of the computational complexity, and the rank adaptation technique. Section VI presents and discusses the simulation results and Section VII gives the concluding remarks.

II. System Model

Let us consider a sensor-array system equipped with a uniform linear array (ULA) of M elements, as shown in Fig. 1. Assuming that the sources are in the far field of the array, the signals of K narrowband sources impinge on the array $(K < M)$ with unknown directions of arrival (DOA) θ_l for $l = 1, 2, \ldots, K$.

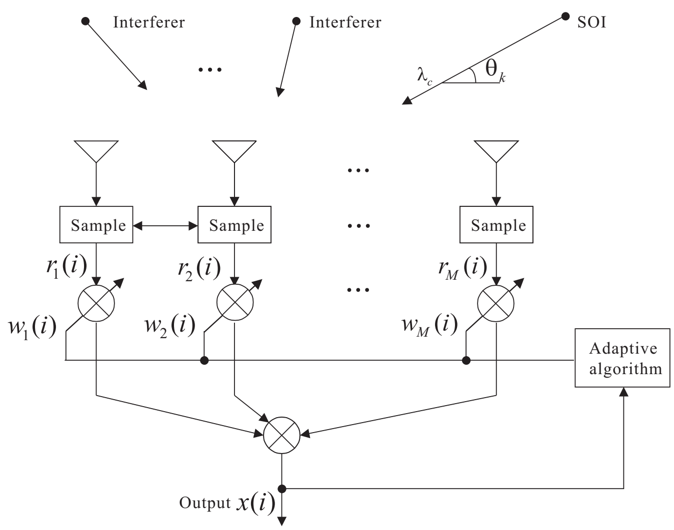

Fig. 1. Block diagram of a sensor-array array system with interfering signals

The input data from the antenna array can be organized in an $M \times 1$ vector expressed by

$$\boldsymbol{r}(i) = \boldsymbol{A}(\theta)\boldsymbol{s}(i) + \boldsymbol{n}(i) \tag{1}$$

where

$$\boldsymbol{A}(\theta) = [\boldsymbol{a}(\theta_1), \ldots, \boldsymbol{a}(\theta_K)]$$

is the $M \times K$ matrix of signal steering vectors. The $M \times 1$ signal steering vector is defined as

$$\boldsymbol{a}(\theta_l) = \left[1, e^{-2\pi j \frac{d_s}{\lambda_c}\cos\theta_l}, \ldots, e^{-2\pi j (M-1)\frac{d_s}{\lambda_c}\cos\theta_l}\right]^T \tag{2}$$

for a signal impinging at angle θ_l, $l = 1, 2, \ldots, K$, where $d_s = \lambda_c/2$ is the inter-element spacing, λ_c is the wavelength and $(.)^T$ denotes the transpose operation. The vector $\boldsymbol{n}(i)$ denotes the complex vector of sensor noise, which is assumed to be zero-mean and Gaussian with covariance matrix $\sigma^2\boldsymbol{I}$.

III. Problem Statement and Design of Adaptive Beamformers

In this section, the problem of designing algorithms against steering vector mismatches is stated. The design of robust full-rank and reduced-rank LCMV beamformers is introduced along with the modelling of steering vector mismatches. The presumed array steering vector for the k-th desired signal is given by $\boldsymbol{a}_p(\theta_k) = \boldsymbol{a}(\theta_k) + \boldsymbol{e}$, where $\boldsymbol{e}$ is the $M \times 1$ mismatch vector and $\boldsymbol{a}(\theta_k)$ is the actual array steering vector which is unknown by the system. By using the presumed array steering vector $\boldsymbol{a}_p(\theta_k)$, the performance of a conventional LCMV beamformer can be degraded significantly. The problem of interest is how to design a beamformer that can deal with the mismatch and minimize the performance loss due to the uncertainty.

A. Adaptive LCMV Beamformers

In order to perform beamforming with a full-rank LCMV beamformer, we linearly combine the data vector $\boldsymbol{r}(i) = [r_1^{(i)}\ r_2^{(i)}\ \ldots\ r_M^{(i)}]^T$ with the full-rank beamformer $\boldsymbol{w} = [w_1\ w_2\ \ldots\ w_M]^T$ to yield

$$x(i) = \boldsymbol{w}^H\boldsymbol{r}(i) \tag{3}$$

The optimal LCMV beamformer is described by the $M \times 1$ vector $\boldsymbol{w}$, which is designed to solve the following optimization problem

$$\begin{aligned} &\text{minimize } E[|\boldsymbol{w}^H\mathbf{r}(i)|^2] = \boldsymbol{w}^H\boldsymbol{R}\boldsymbol{w} \\ &\text{subject to } \boldsymbol{w}^H\boldsymbol{a}(\theta_k)\ = 1 \end{aligned} \tag{4}$$

The solution to the problem in (4) is given by [3], [4]

$$\boldsymbol{w}_{\text{opt}} = \frac{\boldsymbol{R}^{-1}\boldsymbol{a}(\theta_k)}{\boldsymbol{a}^H(\theta_k)\boldsymbol{R}^{-1}\boldsymbol{a}(\theta_k))}, \tag{5}$$

where $\boldsymbol{a}(\theta_k)$ is the steering vector of the SoI, $\boldsymbol{r}(i)$ is the received data, the covariance matrix of $\boldsymbol{r}(i)$ is described by $\boldsymbol{R} = E[\boldsymbol{r}(i)\boldsymbol{r}^H(i)]$, $(\cdot)^H$ denotes Hermitian transpose and $E[\cdot]$ stands for expected value. The beamformer $\boldsymbol{w}(i)$ can be estimated via SG or RLS algorithms [3]. However, the laws that govern their convergence and tracking behaviors imply that they depend on M and on the eigenvalue spread of $\boldsymbol{R}$.

A reduced-rank algorithm must extract the most important features of the processed data by performing dimensionality reduction. This mapping is carried out by a $M \times D$ rank-reduction matrix $\boldsymbol{S}_D$ on the received data as given by

$$\bar{\boldsymbol{r}}(i) = \boldsymbol{S}_D^H\boldsymbol{r}(i) \tag{6}$$

where, in what follows, all D-dimensional quantities are denoted with a "bar". The resulting projected received vector $\bar{\boldsymbol{r}}(i)$ is the input to a beamformer represented by the D vector $\bar{\boldsymbol{w}} = [\bar{w}_1\ \bar{w}_2\ \ldots \bar{w}_D]^T$. The filter output is

$$\bar{x}(i) = \bar{\boldsymbol{w}}^H\bar{\boldsymbol{r}}(i) \tag{7}$$

In order to design a reduced-rank beamformer $\bar{\boldsymbol{w}}$ we consider the following optimization problem

$$\begin{aligned} &\text{minimize } E\left[|\bar{\boldsymbol{w}}^H\bar{\boldsymbol{r}}(i)|^2\right] = \bar{\boldsymbol{w}}^H\bar{\boldsymbol{R}}\bar{\boldsymbol{w}} \\ &\text{subject to } \bar{\boldsymbol{w}}^H\bar{\boldsymbol{a}}(\theta_k) = 1 \end{aligned} \tag{8}$$

The solution to the above problem is

$$\begin{aligned} \bar{\boldsymbol{w}}_{\text{opt}} &= \frac{\bar{\boldsymbol{R}}^{-1}\bar{\boldsymbol{a}}(\theta_k)}{\bar{\boldsymbol{a}}^H(\theta_k)\bar{\boldsymbol{R}}^{-1}\bar{\boldsymbol{a}}(\theta_k)} \\ &= \frac{(\boldsymbol{S}_D^H\boldsymbol{R}\boldsymbol{S}_D)^{-1}\boldsymbol{S}_D^H\boldsymbol{a}(\theta_k)}{\boldsymbol{a}^H\boldsymbol{S}_D(\theta_k)(\boldsymbol{S}_D^H\boldsymbol{R}\boldsymbol{S}_D)^{-1}\boldsymbol{S}_D^H\boldsymbol{a}(\theta_k)}, \end{aligned} \tag{9}$$

where the reduced-rank covariance matrix is $\bar{\boldsymbol{R}} = E[\bar{\boldsymbol{r}}(i)\bar{\boldsymbol{r}}^H(i)] = \boldsymbol{S}_D^H\boldsymbol{R}\boldsymbol{S}_D$ and the reduced-rank steering vector is $\bar{\boldsymbol{a}}(\theta_k) = \boldsymbol{S}_D^H\boldsymbol{a}(\theta_k)$. The above development shows that the choice of $\boldsymbol{S}_D$ to perform dimensionality reduction on $\boldsymbol{r}(i)$ is very important, and can lead to an improved convergence and tracking performance over the full-rank beamformer. A key problem with the full-rank and the reduced-rank beamformers described in (5) and (9), respectively, is that their performance is deteriorated when they employ the presumed array steering vector $\boldsymbol{a}_p(\theta_k)$. In these situations it is fundamental to employ a robust technique that can mitigate the effects of the mismatches between the actual and the presumed steering vector.

B. Robust Adaptive LCMV Beamformers

An effective technique for is the use of diagonal loading strategies [8], [10], [13], [14]. In what follows, robust full-rank and reduced-rank LCMV beamforming designs are described. A general approach based on diagonal loading is employed for both full-rank and reduced-rank designs.

A robust full-rank LCMV beamformer represented by an $M \times 1$ vector $\boldsymbol{w}$ can be designed by solving the following optimization problem

$$\begin{aligned} \text{minimize } & E[|\boldsymbol{w}^H\mathbf{r}(i)|^2] + \epsilon^2||\boldsymbol{w}||^2 = \boldsymbol{w}^H\boldsymbol{R}\boldsymbol{w} + \epsilon^2\boldsymbol{w}^H\boldsymbol{w} \\ & \text{subject to } \boldsymbol{w}^H\boldsymbol{a}(\theta_k) \ = 1, \end{aligned} \tag{10}$$

where ϵ^2 is constant that needs to be chosen by the designer. The solution to the problem in (10) is given by

$$\boldsymbol{w}_{\text{opt}} = \frac{(\boldsymbol{R} + \epsilon^2\boldsymbol{I}_M)^{-1}\boldsymbol{a}_p(\theta_k)}{\boldsymbol{a}_p^H(\theta_k)(\boldsymbol{R} + \epsilon^2\boldsymbol{I}_M)^{-1}\boldsymbol{a}_p(\theta_k))} \tag{11}$$

where $\boldsymbol{a}_p(\theta_k)$ is the presumed steering vector of the SoI and $\boldsymbol{I}_D$ is an M-dimensional identity matrix. It turns out that the adjustment of ϵ^2 needs to be obtained numerically by an optimization algorithm.

In order to design a robust reduced-rank LCMV beamformer $\bar{\boldsymbol{w}}$ we follow a similar approach to the full-rank case and consider the following optimization problem

$$\begin{aligned} \text{minimize } E\big[|\bar{\boldsymbol{w}}^H\boldsymbol{S}_D^H\boldsymbol{r}(i)|^2\big] + \epsilon^2||\boldsymbol{S}_D\bar{\boldsymbol{w}}||^2 &= \bar{\boldsymbol{w}}^H\boldsymbol{S}_D^H\boldsymbol{R}\boldsymbol{S}_D\bar{\boldsymbol{w}} \\ &+ \epsilon^2\bar{\boldsymbol{w}}^H\boldsymbol{S}_D^H\boldsymbol{S}_D\bar{\boldsymbol{w}} \\ \text{subject to } \bar{\boldsymbol{w}}^H\boldsymbol{S}_D^H\boldsymbol{a}_p(\theta_k) = 1, & \end{aligned} \tag{12}$$

The solution to the above problem is

$$\bar{\boldsymbol{w}}_{\text{opt}} = \frac{(\boldsymbol{S}_D^H\boldsymbol{R}\boldsymbol{S}_D + \epsilon^2\boldsymbol{I}_D)^{-1}\boldsymbol{S}_D^H\boldsymbol{a}_p(\theta_k)}{\boldsymbol{a}_p^H\boldsymbol{S}_D(\theta_k)(\boldsymbol{S}_D^H\boldsymbol{R}\boldsymbol{S}_D + \epsilon^2\boldsymbol{I}_D)^{-1}\boldsymbol{S}_D^H\boldsymbol{a}_p(\theta_k)} \tag{13}$$

where the tuning of ϵ^2 requires an algorithmic approach as there is no closed-form solution and $\boldsymbol{I}_D$ is a D-dimensional identity matrix.

IV. Robust Reduced-Rank Beamforming Based on Joint Iterative Optimization of Parameters

In this section, the principles of the robust reduced-rank beamforming scheme based on joint iterative optimization of parameters, termed RJIO, is introduced. The RJIO scheme, depicted in Fig. 2, employs a rank-reduction matrix $\boldsymbol{S}_D(i)$ with dimensions $M \times D$ to perform dimensionality reduction on a data vector $\boldsymbol{r}(i)$ with dimensions $M \times 1$. The reduced-rank beamformer $\bar{\boldsymbol{w}}(i)$ with dimensions $D \times 1$ processes the reduced-rank data vector $\bar{\boldsymbol{r}}(i)$ in order to yield a scalar estimate $\bar{x}(i)$. The rank-reduction matrix $\mathbf{S}_D(i)$ and the reduced-rank beamformer $\bar{\boldsymbol{w}}(i)$ are jointly optimized in the RJIO scheme according to the MV criterion subject to a robust constraint that ensures that the beamforming algorithm is robust against steering vector mismatches and short data records.

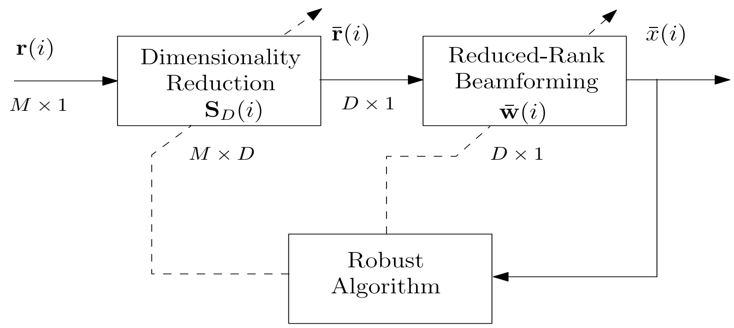

Fig. 2. Block diagram of the RJIO scheme

In order to describe the RJIO method, let us first consider the structure of the $M \times D$ rank-reduction matrix

$$\boldsymbol{S}_D(i) = [\ \boldsymbol{s}_1(i)\ |\ \boldsymbol{s}_2(i)\ |\ \ldots\ |\boldsymbol{s}_D(i)\ ] \tag{14}$$

where the columns $\boldsymbol{s}_d(i)$ for $d = 1, \ \ldots, \ D$ constitute a bank of D robust beamformers with dimensions $M \times 1$ as given by

$$\boldsymbol{s}_d(i) = [s_{1,d}(i)\ s_{2,d}(i)\ \ \ldots\ \ s_{M,d}(i)]^T$$

The output $\bar{x}(i)$ of the RJIO scheme can be expressed as a function of the input vector $\mathbf{r}(i)$, the matrix $\mathbf{S}_D(i)$ and the reduced-rank beamformer $\bar{\mathbf{w}}(i)$:

$$\bar{x}(i) = \bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{r}(i) = \bar{\boldsymbol{w}}^H(i)\bar{\boldsymbol{r}}(i) \tag{15}$$

It is interesting to note that for $D = 1$, the RJIO scheme becomes a robust full-rank LCMV beamforming scheme with an addition weight parameter w_D that provides an amplitude gain. For $D > 1$, the signal processing tasks are changed and the robust full-rank LCMV beamformers compute a subspace projection and the reduced-rank beamformer provides a unity gain in the direction of the SoI. This rationale is fundamental to the exploitation of the low-rank nature of signals in typical beamforming scenarios.

The robust LCMV expressions for $\boldsymbol{S}_D(i)$ and $\bar{\boldsymbol{w}}(i)$ can be computed via the following optimization problem

$$\begin{aligned} \text{minimize } & E\big[|\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{r}(i)|^2\big] + \epsilon^2||\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i)||^2 = \\ & \bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{R}\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i) + \epsilon^2\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i) \\ \text{subject to } & \bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{a}_p(\theta_k) = 1 \end{aligned} \tag{16}$$

In order to solve the above problem, we resort to the method of Lagrange multipliers [3] and transform the into an unconstrained one expressed by the Lagrangian

$$\begin{aligned}\mathcal{L}(\boldsymbol{S}_D(i), \bar{\boldsymbol{w}}(i), \epsilon^2(i)) = E\big[|\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{r}(i)|^2\big] \\ + \epsilon^2(i)\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i)) \\ + [\lambda(\bar{\boldsymbol{w}}^H(i)\mathbf{S}_D^H(i)\boldsymbol{a}_p(\theta_k) - 1],\end{aligned} \tag{17}$$

where λ is a scalar Lagrange multiplier, $*$ denotes complex conjugate. By fixing $\bar{\boldsymbol{w}}(i)$, minimizing (17) with respect to $\boldsymbol{S}_D(i)$ and solving for λ, we get

$$\boldsymbol{S}_D(i) = \frac{(\boldsymbol{R} + \epsilon^2(i)\boldsymbol{I}_M)^{-1}\boldsymbol{a}_p(\theta_k)\bar{\boldsymbol{w}}^H(i)\bar{\boldsymbol{R}}_{\bar{w}}^{-1}}{\bar{\boldsymbol{w}}^H(i)\bar{\boldsymbol{R}}_{\bar{w}}^{-1}\bar{\boldsymbol{w}}(i)\boldsymbol{a}_p^H(\theta_k)(\boldsymbol{R}(i) + \epsilon^2(i)\boldsymbol{I}_M)^{-1}\boldsymbol{a}_p(\theta_k)}, \tag{18}$$

where $\boldsymbol{R} = E[\boldsymbol{r}(i)\boldsymbol{r}^H(i)]$ and $\bar{\boldsymbol{R}}_{\bar{w}} = E[\bar{\boldsymbol{w}}(i)\bar{\boldsymbol{w}}^H(i)]$. By fixing $\boldsymbol{S}_D(i)$, minimizing (17) with respect to $\bar{\boldsymbol{w}}(i)$ and solving for λ, we arrive at the expression

$$\bar{\boldsymbol{w}}(i) = \frac{(\bar{\boldsymbol{R}}(i) + \epsilon^2(i)\boldsymbol{S}_D^H(i)\boldsymbol{I}_D\boldsymbol{S}_D(i))^{-1}\bar{\boldsymbol{a}}_p(\theta_k)}{\bar{\boldsymbol{a}}_p^H(\theta_k)(\bar{\boldsymbol{R}}(i) + \epsilon^2(i)\boldsymbol{S}_D^H(i)\boldsymbol{I}_D\boldsymbol{S}_D(i))^{-1}\bar{\boldsymbol{a}}_p(\theta_k)}, \tag{19}$$

where $\bar{\boldsymbol{R}}(i) = E[\boldsymbol{S}_D^H(i)\boldsymbol{r}(i)\boldsymbol{r}^H(i)\boldsymbol{S}_D(i)] = E[\bar{\boldsymbol{r}}(i)\bar{\boldsymbol{r}}^H(i)]$, $\bar{\boldsymbol{a}}_p(\theta_k) = \boldsymbol{S}_D^H(i)\boldsymbol{a}_p(\theta_k)$. Note that the filter expressions in (18) and (19) are not closed-form solutions for $\bar{\boldsymbol{w}}(i)$ and $\boldsymbol{S}_D(i)$ since (18) is a function of $\bar{\boldsymbol{w}}(i)$ and (19) depends on $\boldsymbol{S}_D(i)$. Thus, it is necessary to iterate (18) and (19) with initial values to obtain a solution. The key strategy lies in the robust joint optimization of the beamformers. The rank D and the diagonal loading parameter $\epsilon^2(i)$ must be adjusted by the designer to ensure appropriate performance or can be estimated via another algorithm. In the next section, iterative solutions via adaptive algorithms are sought for the robust computation of $\boldsymbol{S}_D(i)$, $\bar{\boldsymbol{w}}(i)$, the diagonal loading $\epsilon(i)$ and the rank adaptation.

V. Adaptive Algorithms

In this section, adaptive SG and RLS versions of the RJIO scheme are developed for an efficient implementation. The important issue of determining the rank of the scheme with an adaptation technique is considered. The computational complexity in arithmetic operations of the RJIO-based algorithms is then detailed.

A. Stochastic Gradient Algorithm

In this part, we present a low-complexity SG adaptive reduced-rank algorithm for an efficient implementation of the RJIO method. The basic idea is to employ an alternating optimization strategy to update $\boldsymbol{S}_D(i)$, $\bar{\boldsymbol{w}}(i)$ and the diagonal loading $\epsilon^2(i)$ By computing the instantaneous gradient terms of (17) with respect to $\boldsymbol{S}_D(i)$, $\bar{\boldsymbol{w}}(i)$ and $\epsilon^2(i)$, we obtain

$$\begin{aligned}\nabla\mathcal{L}_{MV\,\boldsymbol{S}_D^*(i)} &= \bar{x}^*(i)\boldsymbol{r}(i)\bar{\boldsymbol{w}}^H(i) + \epsilon^2(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i)\bar{\boldsymbol{w}}^H(i) + 2\lambda^*\boldsymbol{a}_p(\theta_k)\bar{\boldsymbol{w}}^H(i) \\ \nabla\mathcal{L}_{MV\,\bar{\boldsymbol{w}}^*(i)} &= \bar{x}^*(i)\boldsymbol{S}_D^H(i)\mathbf{r}(i) + \epsilon^2(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i) + 2\lambda^*\boldsymbol{S}_D^H(i)\boldsymbol{a}_p(\theta_k) \\ \nabla\mathcal{L}_{MV\,\epsilon^2(i)} &= 2\epsilon(i)\boldsymbol{w}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i)\end{aligned} \tag{20}$$

By introducing the positive step sizes μ_s, μ_w and μ_ϵ, using the gradient rules $\boldsymbol{S}_D(i+1) = \boldsymbol{S}_D(i) - \mu_s\nabla\mathcal{L}_{MV\,\boldsymbol{S}_D^*(i)}$, $\bar{\boldsymbol{w}}(i+1) = \bar{\boldsymbol{w}}(i) - \mu_w\nabla\mathcal{L}_{MV\,\bar{\boldsymbol{w}}^*(i)}$ and $\epsilon(i+1) = \epsilon(i) - \mu_w\nabla\mathcal{L}_{MV\,\epsilon(i)}$, enforcing the constraint and solving the resulting equations, we obtain

$$\begin{aligned}\boldsymbol{S}_D(i+1) = \boldsymbol{S}_D(i) - \mu_s\big[\bar{x}^*(i)\boldsymbol{r}(i)\bar{\boldsymbol{w}}^H(i) + \epsilon(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i)\bar{\boldsymbol{w}}^H(i) \\ - \big(\boldsymbol{a}_p^H(\theta_k)\boldsymbol{a}_p(\theta_k)\big)^{-1}\boldsymbol{a}_p(\theta_k)\bar{\boldsymbol{w}}^H(i)(\bar{x}^*(i)\boldsymbol{a}_p^H(\theta_k)\boldsymbol{r}(i) + \epsilon(i)\end{aligned} \tag{21}$$

$$\begin{aligned}\bar{\boldsymbol{w}}(i+1) = \bar{\boldsymbol{w}}(i) - \mu_w\big(\bar{x}^*(i)\boldsymbol{S}_D^H(i)\boldsymbol{r}(i) + \epsilon(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i) \\ + (\boldsymbol{a}_p^H(\theta_k)\boldsymbol{a}_p(\theta_k))^{-1}(\bar{x}^*(i)\boldsymbol{r}^H(i)\boldsymbol{S}_D(i)\boldsymbol{S}_D^H(i)\boldsymbol{a}_p(\theta_k) + \epsilon(i)\boldsymbol{u}\end{aligned} \tag{22}$$

$$\epsilon(i+1) = \epsilon(i) - \mu_\epsilon\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i), \tag{23}$$

where $\bar{x}(i) = \bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{r}(i)$. The RJIO scheme trades-off a full-rank beamformer against one rank-reduction matrix $\boldsymbol{S}_D(i)$, one reduced-rank beamformer $\bar{\boldsymbol{w}}(i)$ and one adaptive loading recursion operating in an alternating fashion and exchanging information.

B. Recursive Least Squares Algorithms

Here, an RLS algorithm is devised for an efficient implementation of the RJIO method. To this end, let us first consider the Lagrangian

$$\mathcal{L}_{\mathrm{LS}}(\boldsymbol{S}_D(i), \bar{\boldsymbol{w}}(i), \epsilon(i)) = \sum_{l=1}^{i}\alpha^{i-l}\big|\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{r}(l)\big|^2 + \epsilon^2(i)\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(\tag{24}$$

where α is the forgetting factor chosen as a positive constant close to, but less than 1.

Fixing $\bar{\boldsymbol{w}}(i)$, computing the gradient of (24) with respect to $\boldsymbol{S}_D(i)$, equating the gradient terms to zero and solving for λ, we obtain

$$\boldsymbol{S}_D(i) = \frac{\boldsymbol{P}(i)\boldsymbol{a}_p(\theta_k)\boldsymbol{a}_p^H(\theta_k)\boldsymbol{S}_D(i-1)}{\boldsymbol{a}_p^H(\theta_k)\boldsymbol{P}(i)\boldsymbol{a}_p(\theta_k)} \tag{25}$$

where we defined the inverse covariance matrix $\boldsymbol{P}(i) = \boldsymbol{R}^{-1}(i)$ for convenience of presentation. Employing the matrix inversion lemma [3], we obtain

$$\boldsymbol{k}(i) = \frac{\alpha^{-1}\boldsymbol{P}(i-1)\boldsymbol{r}(i)}{1 + \alpha^{-1}\boldsymbol{r}^H(i)\boldsymbol{P}(i-1)\boldsymbol{r}(i)} \tag{26}$$

$$\boldsymbol{P}(i) = \alpha^{-1}\boldsymbol{P}(i-1) - \alpha^{-1}\boldsymbol{k}(i)\boldsymbol{r}^H(i)\boldsymbol{P}(i-1) + \epsilon^2(i)\boldsymbol{I}_M \tag{27}$$

where $\boldsymbol{k}(i)$ is the $M \times 1$ Kalman gain vector. We set $\boldsymbol{P}(0) = \delta\boldsymbol{I}_M$ to start the recursion of (27), where δ is a positive constant.

Assuming $\boldsymbol{S}_D(i)$ is known and taking the gradient of (24) with respect to $\bar{\boldsymbol{w}}(i)$, equating the terms to a null vector and solving for λ, we obtain the $D \times 1$ reduced-rank beamformer

$$\bar{\boldsymbol{w}}(i) = \frac{\bar{\boldsymbol{P}}(i)\boldsymbol{S}_D^H(i)\boldsymbol{a}_p(\theta_k)}{\boldsymbol{a}_p^H(\theta_k)\boldsymbol{S}_D(i)\bar{\boldsymbol{P}}(i)\boldsymbol{S}_D^H(i)\boldsymbol{a}_p(\theta_k)} \tag{28}$$

where $\bar{\boldsymbol{P}}(i) = \bar{\boldsymbol{R}}^{-1}(i)$ and $\bar{\boldsymbol{R}}(i) = \sum_{l=1}^{i} \alpha^{i-l} \bar{\boldsymbol{r}}(l) \bar{\boldsymbol{r}}^H(l)$ is the reduced-rank input covariance matrix. In order to estimate $\bar{\boldsymbol{P}}(i)$, we use the matrix inversion lemma [3] as follows

$$\bar{\boldsymbol{k}}(i) = \frac{\alpha^{-1} \bar{\boldsymbol{P}}(i-1) \bar{\boldsymbol{r}}(i)}{1 + \alpha^{-1} \bar{\boldsymbol{r}}^H(i) \bar{\boldsymbol{P}}(i-1) \bar{\boldsymbol{r}}(i)} \quad (29)$$

$$\bar{\boldsymbol{P}}(i) = \alpha^{-1} \bar{\boldsymbol{P}}(i-1) - \alpha^{-1} \bar{\boldsymbol{k}}(i) \bar{\boldsymbol{r}}^H(i) \bar{\boldsymbol{P}}(i-1) + \epsilon^2(i) \boldsymbol{I}_D \quad (30)$$

where $\bar{\boldsymbol{k}}(i)$ is the $D \times 1$ reduced-rank gain vector and $\bar{\boldsymbol{P}}(i) = \bar{\boldsymbol{R}}^{-1}(i)$ is referred to as the reduced-rank inverse covariance matrix. Hence, the covariance matrix inversion $\bar{\boldsymbol{R}}^{-1}(i)$ is replaced at each step by the recursive processes (29) and (30) for reducing the complexity. The recursion of (30) is initialized by choosing $\bar{\boldsymbol{P}}(0) = \bar{\delta} \boldsymbol{I}_D$, where $\bar{\delta}$ is a positive constant. The last recursion adjusts the diagonal loading according to the following update equation

$$\epsilon(i+1) = \epsilon(i) - \mu_\epsilon \bar{\boldsymbol{w}}^H(i) \boldsymbol{S}_D^H(i) \boldsymbol{S}_D(i) \bar{\boldsymbol{w}}(i), \quad (31)$$

The RJIO-RLS algorithm trades-off a full-rank beamformer with M coefficients against one matrix recursion to compute $\boldsymbol{S}_D(i)$, given in (25)-(27), one $D \times 1$ reduced-rank adaptive beamformer $\bar{\boldsymbol{w}}(i)$, given in (28)-(30), and one recursion to adjust the diagonal loading described in (31) in an alternating manner and exchanging information.

C. Complexity of RJIO Algorithms

Here, we evaluate the computational complexity of the RJIO and analyzed LCMV algorithms. The complexity expressed in terms of additions and multiplications is depicted in Table I. We can verify that the RJIO-SG algorithm has a complexity that grows linearly with DM, which is about D times higher than the full-rank LCMV-SG algorithm and significantly lower than the remaining techniques. If $D << M$ (as we will see later) then the additional complexity can be acceptable provided the gains in performance justify them. In the case of the RJIO-RLS algorithm the complexity is quadratic with M^2 and D^2. This corresponds to a complexity slightly higher than the one observed for the full-rank LCMV-RLS algorithm, provided D is significantly less than M, and lower than the algorithms WC-SOC [10] and WC-ME [11].

TABLE I
COMPUTATIONAL COMPLEXITY OF LCMV ALGORITHMS

Algorithm	Additions	Multiplications
LCMV-SG [4]	$3M+1$	$3M+2$
LCMV-RLS [7]	$3M^2-2M+3$	$6M^2+2M+2$
RJIO-SG	$3DM+4M$ $+2D-2$	$5DM+2M$ $+5D+2$
RJIO-RLS	$3M^2-M+3$ $+3D^2-7D+3$	$7M^2+3M$ $+7D^2+10D$
SMI [24]	$2/3M^3+3M^2$	$2/3M^3+5M^2$

In order to illustrate the main trends in what concerns the complexity of the proposed and analyzed algorithms, we show in Fig. 3 the complexity in terms of additions and multiplications versus the number of input samples M. The curves indicate that the RJIO-RLS algorithm has a complexity lower than the WC-ME [11] and the WC-SOC [10], whereas it remains at the same level of the full-rank LCMV-RLS algorithm. The RJIO-SG algorithm has a complexity that is situated between the full-rank LCMV-RLS and the full-rank LCMV-SG algorithms.

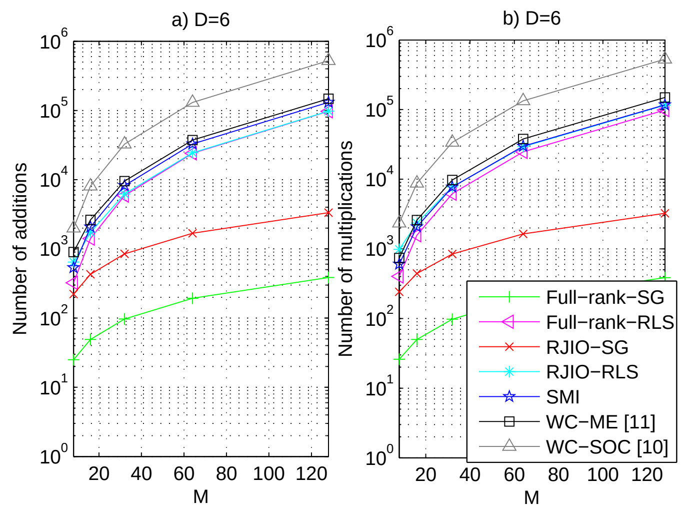

Fig. 3. Computational complexity in terms of arithmetic operations against M

D. Rank Adaptation

The performance of the algorithms described in the previous subsections depends on the rank D. This motivates the development of methods to automatically adjust D on the basis of the cost function. Differently from existing methods for rank adaptation which use MSWF-based algorithms [24] or AVF-based recursions [26], we focus on an approach that jointly determines D based on the LS criterion computed by the filters $\boldsymbol{S}_D(i)$ and $\bar{\boldsymbol{w}}_D(i)$, where the subscript D denotes the rank used for the adaptation. In particular, we present a method for automatically selecting the ranks of the algorithms based on the exponentially weighted *a posteriori* least-squares type cost function described by

$$\mathcal{C}(\boldsymbol{S}_D(i-1), \bar{\boldsymbol{w}}_D(i-1)) = \sum_{l=1}^{i} \alpha^{i-l} \left| \bar{\boldsymbol{w}}_D^H(i-1) \boldsymbol{S}_D(i-1) \boldsymbol{r}(l) \right|^2, \quad (32)$$

where α is the forgetting factor and $\bar{\mathbf{w}}_D(i-1)$ is the beamformer with rank D. For each time interval i, we can select the rank $D_{\rm opt}$ which minimizes $\mathcal{C}(\boldsymbol{S}_D(i-1), \bar{\boldsymbol{w}}_D(i-1))$ and the exponential weighting factor α is required as the optimal rank varies as a function of the data record. The key quantities to be updated are the rank-reduction matrix $\boldsymbol{S}_D(i)$, the beamformer $\bar{\boldsymbol{w}}_D(i)$, the associated presumed reduced-rank steering vector $\bar{\boldsymbol{a}}_p(\theta_k)$ and the inverse of the reduced-rank covariance matrix $\bar{\boldsymbol{P}}(i)$ (for the RJIO-RLS algorithm). To this end, we define the following extended rank-reduction matrix $\boldsymbol{S}_D(i)$ and the extended reduced-rank beamformer weight vector $\bar{\boldsymbol{w}}_D(i)$ as

follows:

$$\boldsymbol{S}_D(i) = \begin{bmatrix} s_{1,1} & s_{1,2} & \dots & s_{1,D_{\min}} & \dots & s_{1,D_{\max}} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ s_{M,1} & s_{M,2} & \dots & s_{M,D_{\min}} & \dots & s_{M,D_{\max}} \end{bmatrix} \text{ and } \bar{\boldsymbol{w}}_D(i) = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{D_{\min}} \\ \vdots \\ w_{D_{\max}} \end{bmatrix} \tag{33}$$

The extended rank-reduction matrix $\boldsymbol{S}_D(i)$ and the extended reduced-rank beamformer weight vector $\bar{\boldsymbol{w}}_D(i)$ are updated along with the associated quantities $\bar{\boldsymbol{a}}(\theta_k)$ and $\bar{\boldsymbol{P}}(i)$ (only for the RLS) for the maximum allowed rank $D_{\max}$ and then the rank adaptation algorithm determines the rank that is best for each time instant i using the cost function in (32). The rank adaptation algorithm is then given by

$$D_{\rm opt} = \arg \min_{D_{\min} \leq d \leq D_{\max}} \mathcal{C}(\boldsymbol{S}_D(i-1), \bar{\boldsymbol{w}}_D(i-1)) \tag{34}$$

where d is an integer, $D_{\min}$ and $D_{\max}$ are the minimum and maximum ranks allowed for the reduced-rank beamformer, respectively. Note that a smaller rank may provide faster adaptation during the initial stages of the estimation procedure and a greater rank usually yields a better steady-state performance. Our studies reveal that the range for which the rank D of the proposed algorithms have a positive impact on the performance of the algorithms is limited, being from $D_{\min} = 3$ to $D_{\max} = 8$ for the reduced-rank beamformer recursions. These values are rather insensitive to the system load (number of users), to the number of array elements and work very well for all scenarios and algorithms examined. The additional complexity of the proposed rank adaptation algorithm is that it requires the update of all involved quantities with the maximum allowed rank $D_{\max}$ and the computation of the cost function in (32). This procedure can significantly improve the convergence performance and can be relaxed (the rank can be made fixed) once the algorithm reaches steady state. Choosing an inadequate rank for adaptation may lead to performance degradation, which gradually increases as the adaptation rank deviates from the optimal rank.

VI. Simulations

In this section, the performance of the RJIO and some existing beamforming algorithms is assessed using computer simulations. A sensor-array system with a ULA equipped with M sensor elements is considered for assessing the beamforming algorithms. In particular, the performance of the RJIO scheme with SG and RLS algorithms is compared with existing techniques, namely, the full-rank LCMV-SG [4] and LCMV-RLS [7], and the robust techniques reported in [10], termed WC-SOC, and [11], called Robust-ME, and the optimal linear beamformer that assumes the knowledge of the covariance matrix and the actual steering vector [2]. In particular, the algorithms are compared in terms of the signal-to-interference-plus-noise ratio (SINR), which is defined for the schemes as

$$\text{SINR}(i) = \frac{\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{R}_s\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i)}{\bar{\boldsymbol{w}}^H(i)\boldsymbol{S}_D^H(i)\boldsymbol{R}_I\boldsymbol{S}_D(i)\bar{\boldsymbol{w}}(i)}, \tag{35}$$

where $\boldsymbol{R}_s$ is the covariance matrix of the desired signal and $\boldsymbol{R}_I$ is the covariance matrix of the interference and noise in the environment. Note that for the full-rank schemes the SINR(i) assumes $\boldsymbol{S}_D^H(i) = \boldsymbol{I}_M$. For each scenario, 200 runs are used to obtain the curves. In all simulations, the desired signal power is $\sigma_d^2 = 1$, and the signal-to-noise ratio (SNR) is defined as $\text{SNR} = \frac{\sigma_d^2}{\sigma^2}$. The beamformers are initialized as $\bar{\boldsymbol{w}}(0) = [1 \; 0 \dots 0]$ and $\boldsymbol{S}_D(0) = [\boldsymbol{I}_D^T \; \boldsymbol{0}_{D\times(M-D)}^T]$, where $\boldsymbol{0}_{D\times M-D}$ is a $D \times (M-D)$ matrix with zeros in all experiments.

In order to assess the performance of the RJIO and other existing algorithms in the presence of uncertainties, we consider that the array steering vector is corrupted by local coherent scattering

$$\boldsymbol{a}_p(\theta_k) = \boldsymbol{a}(\theta_k) + \sum_{k=1}^{4} e^{j\Phi_k}\boldsymbol{a}_{\rm sc}(\theta_k), \tag{36}$$

where Φ_k is uniformly distributed between zero and 2π and θ_k is uniformly distributed with a standard deviation of 2 degrees with the assumed direction as the mean. The mismatch changes for every realization and is fixed over the snapshots of each simulation trial. In the first two experiments, we consider a scenario with 7 interferers at -60^o, -45^o, 30^o -15^o, 0^o, 45^o, 60^o with powers following a log-normal distribution with associated standard deviation 3 dB around the SoI's power level. The SoI impinges on the array at 30^o. The parameters of the algorithms are optimized.

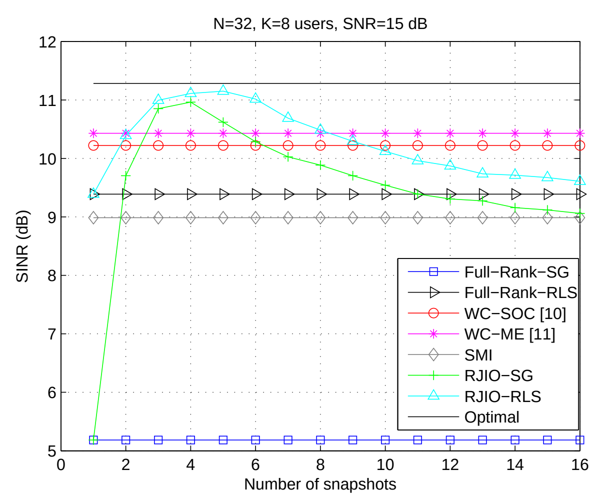

Fig. 4. SINR performance of LCMV algorithms against rank (D) with $M = 32$, $SNR = 15$ dB, $N = 250$ snapshots

We first evaluate the SINR performance of the analyzed algorithms against the rank D using optimized parameters (μ_s, μ_w and forgetting factors λ) for all schemes and $N = 250$ snapshots. The results in Fig. 4 indicate that the best rank for the RJIO scheme is $D = 4$ (which will be used in the second scenario) and it is very close to the optimal full-rank LCMV beamformer that has knowledge about the actual steering vector. An examination of systems with different sizes has shown that D is relatively invariant to the system size,

which brings considerable computational savings. In practice, the rank D can be adapted in order to obtain fast convergence and ensure good steady-state performance and tracking after convergence.

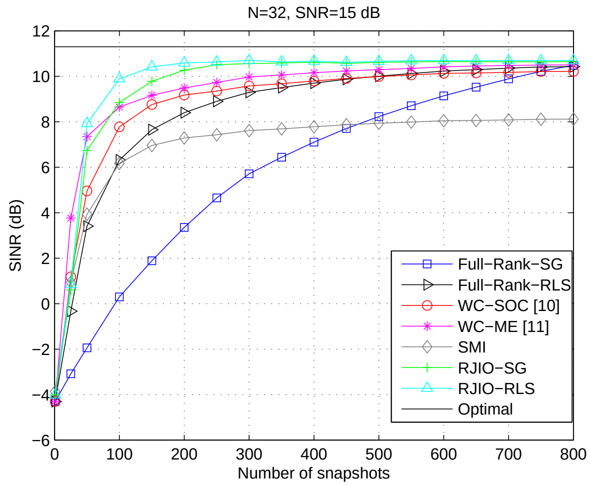

Fig. 5. SINR performance of robust LCMV algorithms against snapshots with $M = 32$, $SNR = 15$ dB

We display another scenario in Fig. 5 where the robust adaptive LCMV beamformers are set to converge to the same level of SINR. The parameters used to obtain these curves are also shown. The curves show an excellent performance for the RJIO scheme which converges much faster than the full-rank-SG algorithm, and is also better than the more complex WC-SOC [10] and Robust-ME [11] schemes.

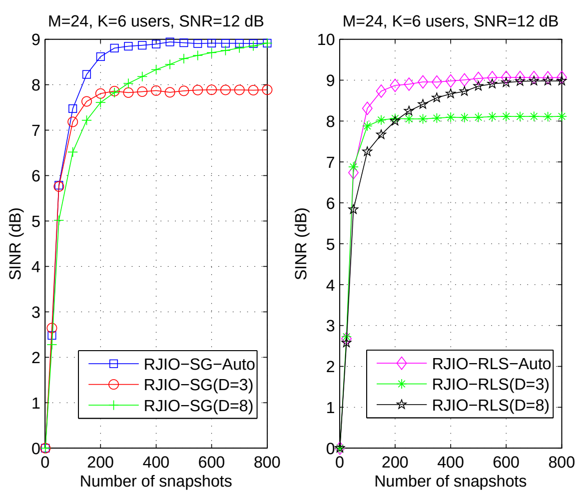

Fig. 6. SINR performance of RJIO-LCMV (a) SG and (b) RLS algorithms against snapshots with $M = 24$, $SNR = 12$ dB with rank adaptation

In the next example, we consider the design of the RJIO-SG and RJIO-RLS algorithms equipped with the rank adaptation method described in Section V.D. We consider 5 interferers at -60^o, -30^o, 0^o, 45^o, 60^o with equal powers to the SoI, which impinges on the array at 15^o. Specifically, we evaluate the rank adaptation algorithms against the use of fixed ranks, namely, $D = 3$ and $D = 8$ for both SG and RLS algorithms. The results show that the rank adaptation method is capable of ensuring an excellent trade-off between convergence speed and steady-state performance, as illustrated in Fig 6. In particular, the algorithm can achieve a significantly faster convergence performance than the scheme with fixed rank $D = 8$, whereas it attains the same steady state performance.

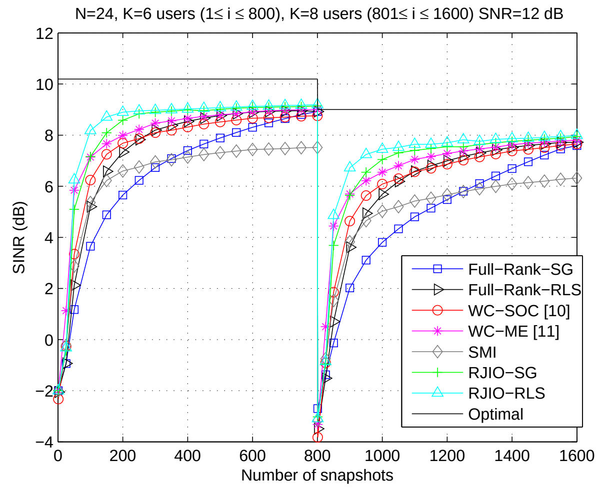

Fig. 7. SINR performance of robust LCMV algorithms against the number of snapshots with $M = 24$, $SNR = 12$ dB in a non-stationary scenario

In the last experiment, we consider a non-stationary scenario where the system has 6 users with equal power and the environment experiences a sudden change at time $i = 800$. The 5 interferers impinge on the ULA at -60^o, -30^o, 0^o, 45^o, 60^o with equal powers to the SoI, which impinges on the array at 15^o. At time instant $i = 800$ we have 3 interferers with 5 dB above the SoI's power level entering the system with DoAs -45^o, -15^o and 30^o, whereas one interferer with DoA 45^o and a power level equal to the SoI exits the system. The RJIO and other analyzed algorithms are equipped with rank adaptation techniques and have to adjust their parameters in order to suppress the interferers. We optimize the step sizes and the forgetting factors of all the algorithms in order to ensure that they converge as fast as they can to the same value of SINR. The results of this experiment are depicted in Fig. 7. The curves show that the RJIO algorithms have a superior performance to the existing algorithms considered in this study.

VII. Conclusions

We have investigated robust reduced-rank LCMV beamforming algorithms based on robust joint iterative optimization of beamformers. The RJIO reduced-rank scheme is based on a robust constrained joint iterative optimization of beamformers according to the minimum variance criterion. We derived robust LCMV expressions for the design of the rank-reduction matrix and the reduced-rank beamformer and developed SG and RLS adaptive algorithms for their efficient implementation along with a rank adaptation technique. The numerical results

for an application with a ULA have shown that the RJIO scheme and algorithms outperform in convergence, steady state and tracking the existing robust full-rank and reduced-rank algorithms at comparable complexity. The proposed algorithms can be extended to other array geometries and applications.

REFERENCES

[1] H. L. Van Trees, "Detection, Estimation, and Modulation Theory, Part IV, Optimum Array Processing", John Wiley & Sons, 2002.

[2] J. Li and P. Stoica, "Robust adaptive Beamforming", *Wiley* , 2006.

[3] S. Haykin *Adaptive Filter Theory*, Prentice Hall, 4rd ed, 2002.

[4] O. L. Frost III, "An algorithm for linearly constrained adaptive array processing" *Proceedings of IEEE*, vol. AP-30, pp. 27 - 34, Jan 1972.

[5] R. C. de Lamare and R. Sampaio-Neto, "Low-Complexity Variable Step-Size Mechanisms for Stochastic Gradient Algorithms in Minimum Variance CDMA Receivers", IEEE Trans. Signal Processing, vol. 54, pp. 2302 - 2317, June 2006.

[6] B. D. Van Veen, "Adaptive covergence of linearly constrained beamformers based on the sample covariance matrix", *IEEE Transactions on Signal Processing*, vol. 39, pp. 1470-1473, 1991.

[7] L. S. Resende, J. M. T. Romano and M. G. Bellanger "A fast least-squares algoirthm for linearly constrained adaptive filtering" *IEEE Transactions on Signal Processing*, vol. 44, pp. 1168 - 1174, May 1996.

[8] H. Cox, R. M. Zeskind, and M. H. Owen, "Robust adaptive beamforming," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, pp. 1365-1376, Oct. 1987.

[9] D. D. Feldman and L. J. Griffiths, "A projection approach for robust adaptive beamforming", *IEEE Transactions on Signal Processing*, vol. 42, pp. 867-876, 1994.

[10] S. A. Vorobyov, A. B. Gershman, and Z.-Q. Luo, "Robust adaptive beamforming using worst-case performance optimization: A solution to the signal mismatch problem," *IEEE Trans. Signal Processing*, vol. 51, pp. 313-324, Feb. 2003.

[11] J. Li and P. Stoica, "On robust capon beamforming and diagonal loading," *IEEE Trans. Signal Processing*, vol. 51, no. 7, pp. 1702-1715, Jul. 2003.

[12] P. Stoica, J. Li and Z. Wang, "Doubly constrained robust capon beamformer," *IEEE Transactions on Signal Processing*, vol. 52, no. 9, pp. 2407-2423, sept. 2004.

[13] R. Lorenz and S. Boyd, "Robust minimum variance beamforming," *IEEE Trans. Signal Processing*, vol. 53, pp. 1684-1696, May 2005.

[14] C.-Y. Chen and P. P. Vaidyanathan, "Quadratically constrained beamforming robust against direction-of-arrival mismatch," *IEEE Trans. Signal Processing*, vol. 55, no. 8, 2007.

[15] A. Hassanien and S. A. Vorobyov, "Robust adaptive beamforming using sequential quadratic programming: An iterative solution to the mismatch problem," *IEEE Signal Processing Lett.*, vol. 15, pp. 733-736, 2008.

[16] L. Wang and R. C. de Lamare, "Low-Complexity Adaptive Step Size Constrained Constant Modulus SG Algorithms for Blind Adaptive Beamforming", *Signal Processing*, vol. 89, no. 12, December 2009, pp. 2503-2513.

[17] L. Wang, R. C. de Lamare, M. Yukawa, "Adaptive Reduced-Rank Constrained Constant Modulus Algorithms Based on Joint Iterative Optimization of Filters for Beamforming," *IEEE Transactions on Signal Processing*, vol.58, no.6, June 2010, pp.2983-2997.

[18] R.C. de Lamare, R. Sampaio-Neto and M. Haardt, "Blind Adaptive Constrained Constant-Modulus Reduced-Rank Interference Suppression Algorithms Based on Interpolation and Switched Decimation," *IEEE Trans. on Signal Processing*, vol.59, no.2, pp.681-695, Feb. 2011.

[19] L. L. Scharf and D. W. Tufts, "Rank reduction for modeling stationary signals," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-35, pp. 350-355, March 1987.

[20] A. M. Haimovich and Y. Bar-Ness, "An eigenanalysis interference canceler," *IEEE Trans. on Signal Processing*, vol. 39, pp. 76-84, Jan. 1991.

[21] D. A. Pados and S. N. Batalama "Joint space-time auxiliary vector filtering for DS/CDMA systems with antenna arrays" *IEEE Transactions on Communications*, vol. 47, no. 9, pp. 1406 - 1415, 1999.

[22] J. S. Goldstein, I. S. Reed and L. L. Scharf "A multistage representation of the Wiener filter based on orthogonal projections" *IEEE Transactions on Information Theory*, vol. 44, no. 7, 1998.

[23] Y. Hua, M. Nikpour and P. Stoica, "Optimal reduced rank estimation and filtering," IEEE Transactions on Signal Processing, pp. 457-469, Vol. 49, No. 3, March 2001.

[24] M. L. Honig and J. S. Goldstein, "Adaptive reduced-rank interference suppression based on the multistage Wiener filter," *IEEE Transactions on Communications*, vol. 50, no. 6, June 2002.

[25] E. L. Santos and M. D. Zoltowski, "On Low Rank MVDR Beamforming using the Conjugate Gradient Algorithm", *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2004.

[26] Q. Haoli and S.N. Batalama, "Data record-based criteria for the selection of an auxiliary vector estimator of the MMSE/MVDR filter", *IEEE Transactions on Communications*, vol. 51, no. 10, Oct. 2003, pp. 1700 - 1708.

[27] R. C. de Lamare and R. Sampaio-Neto, "Reduced-Rank Adaptive Filtering Based on Joint Iterative Optimization of Adaptive Filters", *IEEE Signal Processing Letters*, Vol. 14, no. 12, December 2007.

[28] R. C. de Lamare, "Adaptive Reduced-Rank LCMV Beamforming Algorithms Based on Joint Iterative Optimisation of Filters", *Electronics Letters*, vol. 44, no. 9, 2008.

[29] R. C. de Lamare and R. Sampaio-Neto, "Adaptive Reduced-Rank Processing Based on Joint and Iterative Interpolation, Decimation and Filtering", *IEEE Transactions on Signal Processing*, vol. 57, no. 7, July 2009, pp. 2503 - 2514.

[30] R. C. de Lamare, L. Wang, and R. Fa, "Adaptive reduced-rank LCMV beamforming algorithms based on joint iterative optimization of filters: Design and analysis," Signal Processing, vol. 90, no. 2, pp. 640-652, Feb. 2010.

[31] R. Fa, R. C. de Lamare, and L. Wang, "Reduced-Rank STAP Schemes for Airborne Radar Based on Switched Joint Interpolation, Decimation and Filtering Algorithm," *IEEE Transactions on Signal Processing*, vol.58, no.8, Aug. 2010, pp.4182-4194.

[32] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, 2nd Ed., 1999.

[33] G. H. Golub and C. F. van Loan, *Matrix Computations*, 3rd ed., The Johns Hopkins University Press, Baltimore, Md, 1996.